A Windows-based roadway maintenance management system

Ranjit Menon

University of Nevada, Las Vegas

Follow this and additional works at: https://digitalscholarship.unlv.edu/rtds

Repository Citation
https://digitalscholarship.unlv.edu/rtds/1102

This Thesis is brought to you for free and open access by Digital Scholarship@UNLV. It has been accepted for inclusion in UNLV Retrospective Theses & Dissertations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
A WINDOWS BASED ROADWAY MAINTENANCE MANAGEMENT SYSTEM

by

Ranjit Menon

Bachelor of Technology
Department of Civil Engineering
Indian Institute of Technology,
Chennai, India
1997

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science in Engineering
Department of Civil and Environmental Engineering

Graduate College
University of Nevada, Las Vegas
December 1999
The Thesis prepared by

Ranjit Menon

Entitled

A Windows Based Roadway Maintenance Management System

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Engineering

Examination Committee Chair

Dean of the Graduate College

Examination Committee Member

Graduate College Faculty Representative
I dedicate this thesis to my parents
ABSTRACT

A Windows Based Roadway Maintenance Management System

By

Ranjit Menon

Dr. Shashi Sathisan Nambisan, P.E.
Professor of Civil Engineering
University of Nevada, Las Vegas

A Maintenance Management System (MMS), as the name suggests, is a system used to manage maintenance activities on any infrastructure system or facility such as a road network. Maintenance of a road network involves repair, construction and improvement of pavements and the right of way elements that deteriorate due to their usage, exposure to the environment and various other causes such as impact during crashes, vandalism etc. Management of the maintenance activities is critical to maintain the level of service of the road network, public safety and welfare and for effective and efficient fiscal planning. Since this involves large expenses, a system that would improve the efficiency is not only useful but also necessary.

The development of a prototype roadway MMS is presented in this thesis. The system is developed using Visual Basic 6.0 (VB) as the programming language, MS-Access (Access) as the database, and MS-Excel for generating reports. This application
runs on a computer that uses Windows NT/98/95 as the operating system. All the information is stored in Access as tables. These tables are queried, analyzed and maintained using VB. The user interface, developed in VB, facilitates learning and using the system. The basic information such as the names of the personnel, street, equipment and material, the wage rates, the lengths of the street segments, the maintenance activities etc. need to be entered into the system before details about projects can be stored. The application has the appropriate functions to allow the user to input such data. The user can obtain a summary report showing the total cost, the breakdown of the cost by personnel, equipment and material, and the amount that is left to be paid for each project. The user also can query the database and also view the different tables. The roadway MMS as developed here helps record all the maintenance activities, details about fiscal resource allocation, and personnel allocation. Such records can be maintained, analyzed, queried, and evaluated at various levels such as project, roadway, roadway segments etc. The use of this system is illustrated using a case study with data obtained from Clark County, Nevada.

Particular attention has been paid to make this system secure and yet allow more than one user to access the database. In particular, an administrator can allow and deny access to users selectively. This roadway MMS is just a prototype and many improvements and changes can be made to make the system more robust. The system developed here provides a foundation for further improvements in the future.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
# TABLE OF CONTENTS

ABSTRACT ............................................................................................................................... iv

TABLE OF CONTENTS .......................................................................................................... vi

LIST OF FIGURES ................................................................................................................ viii

ACKNOWLEDGEMENTS ..................................................................................................... ix

CHAPTER 1 INTRODUCTION ............................................................................................ I
  1.1 Types of Maintenance Activities................................................................................. 2
  1.2 Pavement Management Systems ................................................................................. 3
  1.3 Pavement Markers and Markings ............................................................................... 5
  1.4 Traffic Control Devices ............................................................................................... 6
  1.5 Significance of the Application Developed in this Thesis .............................................. 6

CHAPTER 2 LITERATURE REVIEW ............................................................................... 9
  2.1 General Purpose Computerized Maintenance Management System ...................... 9
  2.2 Pavement Management Systems .............................................................................. 12

CHAPTER 3 SCOPE OF A MAINTENANCE MANAGEMENT SYSTEM ....................... 15
  3.1 Basic Functional Requirements ................................................................................. 15
  3.2 Successful Implementation ......................................................................................... 17
  3.3 Usefulness of the Tool ............................................................................................... 22

CHAPTER 4 ABOUT VISUAL BASIC, MS ACCESS AND MS EXCEL ..................... 24
  4.1 Visual Basic .............................................................................................................. 24
    4.1.1 Introduction ...................................................................................................... 24
    4.1.2 Visual Basic Concepts .................................................................................... 26
    4.1.3 Elements of a Visual Basic Project ............................................................... 27
    4.1.4 Converting a Visual Basic Project into an Executable .................................... 31
    4.1.5 Package and Deployment Wizard ............................................................... 32
  4.2 Microsoft Access ....................................................................................................... 32
  4.3 Microsoft Excel ........................................................................................................ 34
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Design of the Roadway Maintenance Management System</td>
<td>36</td>
</tr>
<tr>
<td>5.1</td>
<td>Brief Overview</td>
<td>36</td>
</tr>
<tr>
<td>5.2</td>
<td>Basic Design</td>
<td>37</td>
</tr>
<tr>
<td>5.3</td>
<td>Functions Available</td>
<td>54</td>
</tr>
<tr>
<td>5.4</td>
<td>Administrative Functions</td>
<td>69</td>
</tr>
<tr>
<td>6</td>
<td>The Maintenance Management System Application</td>
<td>71</td>
</tr>
<tr>
<td>6.1</td>
<td>Hardware and Software Requirements</td>
<td>71</td>
</tr>
<tr>
<td>6.2</td>
<td>Installation</td>
<td>72</td>
</tr>
<tr>
<td>6.3</td>
<td>Using the Application</td>
<td>72</td>
</tr>
<tr>
<td>6.4</td>
<td>Sample Application</td>
<td>72</td>
</tr>
<tr>
<td>6.5</td>
<td>Useful Features of the Application</td>
<td>84</td>
</tr>
<tr>
<td>7</td>
<td>Summary, Conclusions and Recommendations for Future Work</td>
<td>86</td>
</tr>
<tr>
<td>7.1</td>
<td>Summary</td>
<td>86</td>
</tr>
<tr>
<td>7.2</td>
<td>Conclusions</td>
<td>89</td>
</tr>
<tr>
<td>7.3</td>
<td>Recommendations for Future Work</td>
<td>90</td>
</tr>
<tr>
<td>A</td>
<td>A Sample Summary Report from the Clark County Public Works Department Maintenance Management System</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>References</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>Vita</td>
<td>97</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 4.1  ActiveX Data Object Data Control ................................................................... 28
Figure 4.2  Option Button Control .................................................................................. 29
Figure 4.3  Check Box Control ....................................................................................... 30
Figure 4.4  Combo Box ....................................................................................................... 30
Figure 5.1  Flow of Control Flowchart ............................................................................ 38
Figure 5.2  View of the Basic Crew/Personnel Table ......................................................... 56
Figure 5.3  Append/Input Window for Crew/Personnel ...................................................... 58
Figure 5.4  Change/Edit Window for the Main Project Table ........................................... 61
Figure 5.5  Personnel/Material/Equipment Input Form ................................................. 62
Figure 5.6  Analyze/Query Window for the Personnel Table .......................................... 64
Figure 5.7  Summary Report Format Template .................................................................. 67
Figure 5.8  Password Change Window ............................................................................... 69
Figure 5.9  System Administration Window ...................................................................... 70
Figure 6.1  Main Login Screen ......................................................................................... 73
Figure 6.2  Welcome Screen .............................................................................................. 73
Figure 6.3  Task Manager Screen for a General User ......................................................... 74
Figure 6.4  System Administration Screen ....................................................................... 75
Figure 6.5  Completed Main Project Input Form ............................................................... 77
Figure 6.6  Completed Personnel/Material/Equipment Input Form .................................. 78
Figure 6.7  Final Main Input Form .................................................................................... 79
Figure 6.8  Table View of the Project ............................................................................... 80
Figure 6.9  Results of a Query on the Personnel Table .................................................... 81
Figure 6.10 Project Name Selection Window ..................................................................... 82
Figure 6.11 Summary Report ............................................................................................ 83
ACKNOWLEDGEMENTS

I express my deepest appreciation to Dr. Shashi Nambisan for his guidance and help in the completion of this thesis. I thank him for all the useful advice he has provided ever since my arrival in the United States as a graduate student to pursue this master's program in Fall 1997. I really appreciate his patience and understanding and the support he provided to guide me through all the obstacles and challenges I faced during my stay here. It is his insistence on clarity and “attention to detail”, as he would call it, that has helped me hone my writing and communication skills. The persistence to do better and to drive oneself beyond the ordinary are some of his quality traits that I am still trying to imbibe in myself. An extremely knowledgeable and experienced person, Dr. Shashi has taught many of his students and myself how to not just look at graphs and numbers but understand, in his words, “what stories they tell”. I once again express my deepest gratitude and appreciation for the technical knowledge and other skills that he has taught me, which I am sure will hold me in good stead, whatever challenges I face in future.

I also express my gratitude to Dr. Kaseko for the transportation knowledge he has imparted in me through the courses he taught during this program. The courses helped me learn and further my understanding of transportation engineering. I do appreciate his forgiveness and understanding for some of the assignments that were submitted late.

I acknowledge the help and support of Walter Vodrazka Jr., for taking time to review my thesis and suggest many changes that have contributed towards a better report.
I thank my parents and my family for their undying support and encouragement throughout my life. It is the principles, morals and values that they instilled in me that have helped me pull through the difficult times and also savor the joys of success.

I also thank my thesis committee members Dr. Walter Vodrazka and Dr. Kazem Taghva for their patience and effort in reviewing my thesis.

I thank Dr. Srinivas Pulugurtha (a former student of this university) also, for his support and advice ever since I arrived in the United States and for his patience over the last few weeks in waiting till I completed writing my thesis report to go out and party.

Last but not the least, I would like to thank the Department of Civil & Environmental Engineering and the Transportation Research Center for providing me financial aid in the form of a Graduate Assistantship that helped me pay for my education and hence an opportunity to pursue and complete a Master of Science in Engineering degree.
CHAPTER 1

INTRODUCTION

Travel over land has always been one of the key elements in the development of human civilizations and the world in general. The Romans were the first scientific road builders, with the *Via Appia*, or the “Appian Way”, being initiated in 312 B.C. [1]. All the work was hand-placed stone and a lot of attention was given to the wearing surface than smoothness since the roads had to withstand the wear of animal hooves and hard wheels [1].

A maintenance management system (MMS) for roadways as the name suggests is used to manage maintenance activities on a road network. Maintenance activities include repair, construction, rehabilitation and improvement of pavements and the right of way elements such as painted markings and raised pavement markers to delineate the lanes. Other elements of the right of way include traffic control devices, streetlights and street furniture. Section 1.1 gives an overview of the different types of maintenance activities that are carried out on road networks that involve the pavement and also the right of way elements and other facilities such as roadside rest areas, traffic control devices etc. Section 1.2 describes management of pavements in particular and sections 1.3 and 1.4 discuss the issues related to markers and markings, and general work order management
systems. Section 1.5 concludes the chapter with a summary describing the significance of the application developed in this thesis.

1.1 Types of Maintenance Activities

Highway maintenance can be defined as a program to preserve and repair a system of roadways with its elements to its designed or accepted configuration [2]. System elements include travelway surfaces, shoulders, roadsides, drainage facilities, bridges, tunnels, signs, markings, lighting fixtures, truck weighing and inspection facilities etc. [2].

Each of the roadway elements needs maintenance owing to deterioration due to the effects of weather, vegetative growth, traffic wear etc. The travelway surfaces essentially need to be smooth and safe for travel. Restoring material losses, patching, joint or crack filling are some of the physical maintenance activities and removal of snow and ice, replacing and restoring pavement stripes and markings and raised pavement markers are some of the activities related to the right of way elements on the pavement. Similar maintenance is required of the shoulders, roadsides, intersections etc. [2].

The replacement, cleaning and repair of gutters, underdrain, culverts are some of the other maintenance activities. Apart from these the traffic control and service facilities need to be painted, the traffic signals and bulbs need to be maintained regularly to ensure smooth and safe control of traffic flow. Other facilities such as roadside information booths, rest areas etc. also need to be maintained [2].
Such activities and many more comprise the maintenance activities that are typically required on a regular basis to keep the road network safe and functional with an acceptable level of service.

1.2 Pavement Management Systems

Pavement management may be defined as the effective and efficient directing of various activities involved in providing and sustaining pavements in a condition acceptable to the traveling public at the least life cycle cost [3]. It is important to develop a management system since the emphasis these days has changed from expansion to preservation and rehabilitation of existing road networks [3]. Pavement management decisions are based on the judgement and experience of engineers and administrators and on sometimes inconsistent sources of information [3]. There is also a need to keep track of historical information such as the consequences of earlier pavement management decisions in order to learn from the experience [3].

In the late 1960s and early 1970s the term pavement management system began to be used by researchers to describe the entire range of activities involved in providing pavements [1]. At the same time, the initial operational or “working” systems were developed in two major projects. The largest of these was Project 123, conducted by the Texas Highway Department, Texas A&M University and the University of Texas. A series of reports and manuals resulted from this project and it produced many of the modern innovations in pavement analysis. [1]. The other major project was conducted in the National Cooperative Highway Research Program called Project 1-10 [1].
It is difficult to define precisely the dollar value of expenditures in different modes of transport. However, including maintenance as well as new construction, pavements represent approximately half the total highway expenditure [1]. It is important to note that after the initial development of a highway system, expenditures on the pavement system continue to grow as maintenance, rehabilitation and so forth are required [1].

Pavements are of two basic types: rigid and flexible. Rigid pavements are made with Portland concrete cement as the main structural layer and flexible pavements use asphalt concrete for the surface with additional layers below it. A measurement and prediction of pavement performance is very important [4]. After a review of various performance prediction models, it was concluded that an empirical mechanistic model is best suited with a systematic database that includes the structural information, traffic volume and condition rating for each "homogenous" section of the road [4]. For flexible pavements it was found that age is by far the most significant predictor of serviceability and other factors such as traffic volume and weight play a secondary role [4].

Many techniques have been developed to predict pavement damage or distress. Techniques such as by image processing methods [5], or with the help of lasers [6] or by image analysis using fractals [7] are used to detect and predict the deterioration of pavements and to estimate the time when rehabilitation may have to be undertaken.

Roadway deterioration, usually, is not the result of poor design and construction practices but is caused by the inevitable wear and tear that occurs over a period of years [8]. The gradual deterioration of a pavement occurs due to many factors including variations in climate, drainage, soil conditions, traffic loading and truck traffic [8].
Although the function of the pavement varies with the specific user, in modern highway facilities the purpose of the pavement is to serve traffic safely, comfortably and efficiently, at minimum or “reasonable” cost [1].

Pavement Management is generally classified into two categories: the network level and the project level [9]. Pavement Management at the network level deals with summary information related to the entire highway network [9]. This involves programming and policy decisions made by the upper management. On the other hand, pavement management at the project level deals with detailed and technical information related to a specific pavement section [9]. This involves decisions made by middle or lower management [9].

Many agencies use pavement management systems that are built around optimization models of one type or another. But a flexible decision support tools that lets them specify the optimization problems that seem meaningful within the scope of the decision-making process at any given time is needed [10]. Other models have been developed using Genetic Algorithms [11] and also for project level Urban Roadway Management System (URMS) [12]. In order to make the management system easier to use, graphical systems involving Geographic Information Systems (GIS) have been developed [13, 14].

1.3 Pavement Markers and Markings

Pavement markers and markings are used to ensure smooth and safe flow of traffic and to facilitate lane discipline. The markers and markings deteriorate with usage due to damage, traffic wear and tear, weather etc. A GIS-based management system for
pavement markers and markings has already been developed [15, 16]. The GIS system reduces user input and is linked to a series of spreadsheets that are used to calculate the quantities of material used and their costs. However, this system does not keep track of the personnel or crew members and the equipment used for the maintenance activity.

1.4 Traffic Control Devices

A management system for traffic control devices with a GIS interface has already been developed [17, 18]. This system is a graphical user interface between a roadway inventory database and GIS. The roadway inventory database consists of the details about the various traffic control devices such as signs, signals, pavement marking, streetlights etc. This system, like the one mentioned in the previous section about markers and markings, does not take into account the additional costs of crew members and equipment.

1.5 Significance of the Application Developed in this Thesis

The aforementioned systems are only concerned with a certain particular elements of the roadway network. There is a need, therefore, to develop an integrated system that will allow the supervisors and managers keep track of maintenance activities of all the roadway elements. Such an integrated roadway MMS may be used by elected officials and administrators, public works department staff, road maintenance personnel etc. The system proposed in this research will help record, plan and schedule future maintenance activities. All the information about the various activities that are being carried out can be recorded under different project or work order identities or names. Personnel can be
assigned to various projects and this information is also stored along with the project
details. The roadway MMS also provides a way of preparing summary reports about each
project or for a roadway segment or an entire roadway. This can be used to analyze the
amount of money spent on personnel, equipment and materials. Thus, this information
can be used to plan such maintenance activities in future.

A manager can use the integrated roadway MMS so as to make the most efficient
use of the financial, personnel, material and equipment resources available. This system
will also help in keeping track of the work being done and the expenses involved. In turn
this would facilitate historical analyses of maintenance activities and costs by project,
road or personnel. This also will allow the manager to set priorities for various activities
that need to be completed.

The objective of this thesis is to develop a simple, inexpensive yet robust
roadway MMS for use by small and medium sized public works agencies. The system
can be further developed into a Geographic Information System (GIS) based system with
a Graphical User Interface (GUI) in ArcView (a GIS software). This will allow the user
to select the roads graphically from a map and get a visual representation of the location
of various projects and activities being carried out. It was found that ArcView may be
expensive for small offices to purchase and thus it was not further pursued in this thesis.
However such an interface can be developed as part of a bigger package.

A literature review of the currently available Pavement Management Systems and
Maintenance Management Systems in general is presented in Chapter 2. The scope and
functionality of a MMS, in general, is discussed in Chapter 3. A brief review of the
software used in the development of this application namely, Visual Basic, Microsoft
Access, and Microsoft Excel is presented in Chapter 4. The actual design, functionality and usage of this application are described in Chapter 5. A sample application is presented in Chapter 6. Chapter 7 summarizes the report and provides conclusions and recommendations for future work.
CHAPTER 2

LITERATURE REVIEW

A review of the literature revealed that there are many Computerized Maintenance Management Systems (CMMS) commercially available in the market. However, these systems are mainly used for keeping track of maintenance activities carried out in factories and plants. The software is made to ensure smooth functioning of the machines and equipment and schedule maintenance activities to avoid breakdowns. Most of these references were found on the Internet. Four such CMMS are described here highlighting their uses and specifications as mentioned in their respective documentation material. Several CMMS related to roadways also currently exist. Some of the prominent systems from this group are also discussed in this chapter.

2.1 General Purpose CMMS

There are several general purpose CMMS available commercially in the market that serve to help factories and manufacturing plants manage the maintenance activities. A few of these products are described in the following sections.
2.1.1 Maintenance Master

Maintenance Master developed by Norwich Technologies, Ontario, Canada [19], may be used at factories that invest a lot of capital on equipment and facilities. Plant maintenance departments are very busy and hence the product, Maintenance Master has been designed to simplify activities into quick, uncomplicated entries. Virtually every facet of the program is accessible from a single screen and yet it is comprehensive to meet the unique needs of large and small operations. Custom screens and reports ensure that the system will be able to grow with the organizations operations. The Maintenance Master, according to the product specifications, supports a large number of inventory classes, currencies and tax regimes. This makes it a complete set of flexible tools for managing inventory, purchasing and suppliers.

2.1.2 Products from TMA Systems

TMA Systems [20] is a privately held company based in Oklahoma and was incorporated in 1991. One of its products, TMA PowerBase™, provides the capability to easily monitor maintenance costs. With eleven integrated modules, PowerBase manages everything from work orders to inventory to preventive maintenance. TMA AFM-SQL™ is the client/server version of TMA-AFM. This enhances TMA's line of maintenance management software by providing an open, robust and secure enterprise wide solution for total maintenance management. This software allows large volumes of data to be maintained and analyzed. TMA Digital Maintenance Assistant (DMA) is a handheld paperless electronic work order system for the PalmPilot™ Professional and Palm III organizer from 3Com. TMA Systems has also developed a system that can be accessed
and updated over the Internet. This would eliminate the need to develop methods to remotely access the system since the Internet will provide the network. TMA has many other smaller products also.

2.1.3 MaintScape from Advanced Software Designs

Advanced Software Designs (ASD) [21] is an international organization with headquarters in Texas and provide industry leading Computerized Maintenance Management Systems (CMMS) and Instrument/Equipment Calibration Software. One of the main products is called MainScape and it is a modular product offering all the functionality that maintenance departments expect in a CMMS product and all functions required by personnel involved with calibration activities. MaintScape is offered as a single computer application or operating as a client/server system on any popular network. Versions exist for all Microsoft Windows operating environments.

2.1.4 ATLAS line of products from Data-Trak, Inc

Data-Trak [22] incorporated in 1984 as a subsidiary of Freeport Welding, a welding and fabrication shop for the petro-chemical industry. The Atlas line of products deals with Client-Server configurations and supports databases on Oracle 8. It thus provides a scalable database and also features that allow the software to incorporate all the methodologies being followed in the organization without any changes.

The products mentioned above and several others available under the acronym CMMS are mainly used in factories and plants for managing, operating and maintaining equipment and the flow of general plant operations.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
2.2 Pavement Management Systems

There are several products available specifically for pavement management. Some of these software products are provided by public agencies such as Carson City Public Works, American Public Works Association and several transportation research centers at universities while others are available from private agencies. The software developed by private agencies may be much more expensive and may require much more training for implementation.

Some of the public domain software such as MicroPaver developed by American Public Works Association [23], Metropolitan Transportation Commission Pavement Management System (MTC-PMS) [23] developed by the MTC, Oakland, California have databases that are password protected. These also provide advanced features such as generating reports, training classes etc. Other systems such as Carson City Pavement Management System [23] and the Long Beach Pavement Management System [23] developed at the University of Florida do not have password protection and need the database to be purchased separately. Most of the public domain software run on dBase, a database software while a few others run on Microsoft Access and Microsoft Excel. Better customization and additional features are provided by the private domain software. However these may be much more expensive to purchase. Some of the privately developed software are described below.

Visual Pavement Management System (PMS) [23], developed by Texas Research and Development Incorporated is a windows based system with an ArcView GIS interface and other features like graphical summary curves, multiple user access and many features that can be customized by the user. Another software called RoadManager
2000 developed by Vanasse Hangen Brustlin, Inc., has a GIS interface built on AutoCAD automated mapping as well as ArcView.

Many of the privately developed software may cost several thousands of dollars to purchase and the personnel need to be undergo training to use the software efficiently. The software might require the purchase of additional database software such as FoxPro or Xbase that may not be commonly available in administrative offices and may also be expensive.

Many of the systems presently available are concerned with a certain aspect of the roadway maintenance such as markers and markings [15,16] or traffic control devices [17,18]. The application developed in this thesis has a structure that can be used to keep track of the maintenance activities of all the elements of a roadway network.

The application developed in this thesis is a simple and yet useful decision support tool. The software uses MS Access as the database to store the information. MS Access is available with the Microsoft Office Suite that is commonly used in most workplaces and offices. If the user is familiar with MS Access then the database can be used independent of the software to generate customized reports.

The use interface for the application developed here is built with Visual Basic 6.0 (VB). The VB program has been converted into an executable file that can be run on an IBM compatible computer that has a MS-Windows based operating system such as a Windows NT or Windows 95 or Windows 98 and a minimum of 32Mb of memory is recommended. VB allows the entire application to be packaged into one unit that can be distributed on a CD-ROM or on a set of 12, 3.5” floppy disks with 1.44Mb of storage in each. Hence a CD-ROM drive or a floppy drive is required to install the application. This
package will contain all the additional VB files that are needed to run the program and hence the user does not need to purchase VB separately. Once the user runs the setup program, the application downloads all the required files from the CD-ROM and the software is ready for implementation. A printer is will also be needed if summary reports are to be printed.

The application developed in this thesis is structured to help keep track of the three main resources of a maintenance management system: personnel, material, and equipment. Unlike the other software available in the market this application does not perform analysis or apply optimization techniques to schedule the various resources. This application caters to the needs of small and medium sized public works agencies to record, track and analyze costs of the resources on a project. Analysis can also be performed to find out the capital spent on a particular equipment or the cost per lane mile for a project, or the amount of money remaining to be paid etc. This information can be used for planning future work and also for evaluating the work that has been done.

This thesis presents an application that can be used to keep track of maintenance activity on any infrastructure that forms a network. Being cost effective, and simple the application developed in this thesis can be bought and used by many small and medium public works agencies to efficiently keep track and make best use of the available resources.
CHAPTER 3

SCOPE OF A MAINTENANCE MANAGEMENT SYSTEM

A maintenance management system (MMS) should be able to perform certain basic functions. These functions serve the general needs of any agency using this system and help in managing the maintenance activities. The basic functional requirements of a MMS are described in the following section and some advanced features are discussed in subsequent sections.

3.1 Basic Functional Requirements

A main functional requirement of any MMS is that it should be able to maintain a database of the activities carried out by the department concerned. This will help in reducing the paper work required to file details about each activity. This also will help in organizing the management activities. Electronic storage facilitates making changes to the entries and the addition or deletion of records. Any change in a particular record will be reflected throughout the system. For example, an increase in the hourly wage of a crew member needs to be changed at just one place and once saved, the change will be reflected in all the entries that concern that crew member. This will reduce chances of error and will help in easy maintenance of the database. However, such changes need to be monitored closely so as to avoid problems.
Another functional requirement is the provision to calculate or summarize various fields of the database. This is provided in the system as a report preparation functionality. The report prepared provides a summary of the expenditure in each project broken down by personnel, equipment and material costs. This will help the manager evaluate historical data, and plan for the future to allocate resources according to the expenses incurred this time.

A useful functionality is to also view the database elements at any given point of time. This helps in keeping track of what resources are assigned to the different projects and also for how long. This will again help in historical analysis and planning the allocation of equipment and personnel to future projects by estimating the time it would take for the completion of present projects.

The system also helps maintain a record of the crew member details, the material costs and the equipment costs. It also keeps a historical record of all the maintenance projects.

The system should also be able to handle multiple users. This means that the system should be able to allow different users different levels of access (privilege) to the database. This will help improve the database security and at the same time make it useful to people with different levels of privileges.

The basic features are required to make an effective decision support tool. However additional features may make the tool more useful in generating reports and adapting and implementing the system to changing environments and resources. Such features that influence the successful implementation of the system are discussed in the following section.
3.2 Successful Implementation

Most CMMS are capable of delivering the basic functional requirements mentioned in the preceding section. However, the capability of software to meet the functional requirements is still a basic criterion and that doesn't necessarily foretell a system's success. Most systems deliver the minimum functional requirements. Typically software upgrades appear every year, with a major upgrade every three or four years. If the program doesn't quite have a feature that is sought, it probably will appear in the next upgrade, or the next. The ability of software to change with users' needs is what enables CMMS manufacturers to stay in the software business. [19]

One key feature beyond the basic requirements is adaptability. There are three issues involved in adaptability [19]. They are:

- The need to modify the program to meet required and desired functional requirements.
- The special features that make the system adapt exceptionally well to a particular system
- Integration with other corporate systems of information management.

The third issue is important to keep abreast of the improvements and developments in technology and software. The term for this interoperability between software is called Computer Integrated Facilities Management (CIFM). Although there is the technology to implement such a system, there is a lack of industry standards at present and these are being developed [19].
Other issues that lead to successful implementation of a CMMS deal with maintenance information that may be user defined, user originated, and user maintained [20].

User defined: This activity permits users of computerized maintenance management system to define their specific requirements. Users must define what data they require, which reports they require, and how their specific applications are supposed to function within the maintenance organization. While users are developing this information, it becomes very apparent that some educational programs are required to train the users how to manage their maintenance activities within the computerized maintenance management system environment [20].

User originated: This activity puts maintenance management data into the computerized maintenance management system. Each functional area within the maintenance organization is responsible for entering their own data into the system. Educational programs may be required to support the development of procedures required for data input [20].

User maintained: This activity maintains computerized maintenance management system data. Authorized users should be able to update, add and delete the data as required [20]. This keeps the data updated. The program handles updating the data internally, such as the changes to the cost table when the hourly wage for a worker changes, and the user need not be concerned with such operations.

Another key to any successful and seamless integration is planning. Two issues that come up while trying to integrate a CMMS are Enterprise Resource Planning (ERP) and Resource Utilization.
Enterprise Resource Planning ensures a seamless integration between production and maintenance. Thus, the design and detail of both the production plan and the production schedule are critical to a seamless integration. The production plan must contain critical maintenance functions that require completion. The production schedule should include the allocation of the common resources between maintenance and production along with the planning of equipment downtime [21].

Resource Utilization deals with the correct use of the identified resources providing the environment in which the resources are in the correct place when needed and are inventoried at a level where excess funds are not tied up in idle inventory. The resources comprise every type of resource needed to complete the assigned task. The availability of these resources directly impacts a number of groups such as purchasing, personnel, physical facilities, and funds. An organization should provide resource capacity and availability along with coordination of maintenance schedules with production to improve resource utilization and performance at the time of the CMMS integration [21].

Successful setup of an MMS does not necessarily imply a successful future for the system at the organization. The MMS must be built so that it can be changed to suit the changes in the organization and the resources available. This calls for a Continuous Implementation Methodology [22].

The implementation of a MMS is a continuous process. The business may be constantly changing over time, adopting new policies, procedures and methods. Hence the system needs to be continuously re-evaluated for processes and requirements, and
then those changes should be mapped to the MMS. This ensures that the MMS can continue to be used to its fullest potential as the business changes [22].

The success of a MMS also depends on the design and planning of the database. A well-planned strategy for collecting and entering data or converting data from an existing system for populating the new CMMS is a critical part of the implementation [23].

A well-designed data collection form is a valuable tool for collecting the data. The form should contain fields for all information needed in the MMS, as well as an area for comments. An important item to include on the data collection sheet is the structure or field size that the MMS allows for each entry [23]. The data collected should also be checked for missing or incomplete data.

Some fundamental data may have to be entered by typing or importing each item. However it is a good idea to have drop-down menus for secondary data entry. This will reduce the amount of typing that needs to be done and hence will reduce the occurrence of wrong information due to human data entry error. The drop-down menus may contain values such as the designation of the crew member that may be common to many entries. Hence drop-down menus help to build a good database.

A MMS is should also have extensive reporting capabilities. A MMS is only as good as the information that can be retrieved from it. The MMS should provide extensive management reporting capabilities that include detailed and summary reports, graphical reports, and easy to use report-writing tools that do not require programming knowledge [24].
One method to improve the working of the MMS is to make the process of retrieval of data as quick as possible. Removing obsolete data can do this. These data may be archived for historical purposes but removing this data may cause the MMS to function faster [25].

The MMS may be made more accessible and easier to operate and implement if the data communication can be made wireless. With computers getting smaller, lighter and portable, wireless connectivity is becoming very common. There may be connectivity either within a single plant or across multiple plant locations. In-plant wireless local area networks use either radio frequency or infrared communications. Wide-area networks, that offer wireless links across town or across the county, on the other hand, tend to use public radio frequency (or on rare occasions, satellite) communications [26]. All this allows remote access to the database and the system allowing frequent and convenient updates and changes.

Connecting the system to the Internet may extend the remote access feature of the MMS further. For example this would allow the supervisors to get information from the vendors about the availability of the material such as asphalt concrete that may be used for repairing the pavement. Hence the project or work order may be scheduled such that it is coordinated with the vendors schedule thus making the process more efficient [27].

Training the personnel is another key ingredient in the successful implementation of the MMS. The training should be complete and explain all the features of the MMS in detail and not just the basic features. Menu driven MMS may create an impression that the system is user friendly and hence not much training is needed. However a good training with hands on exercises might save a lot of extra work in future [28].
A management system that has several of these features can be used to improve the efficiency and management of the resources available. The application developed here has several of the features and the usefulness of the tool is discussed in the next section.

3.3 Usefulness of the Tool

The application developed in this thesis is an inexpensive decision support tool that can be purchased and used by small city, town or county offices for keeping track of their projects and resource allocation. This application has the basic features to manage a database such as entering new records, deleting or changing existing records, viewing the database etc.

The database stores all the information that may be needed to keep track of the various projects that may be going on at a particular time. The database also stores historical data that can be used for planning for future projects.

Additional features like simple query and analysis help in proper resource allocation and obtaining information about on-going projects. There is also the provision to generate summary reports that give the final cost of the project and also the individual cost for each resource. The amount that has already been paid and the balance left to be paid are also provided in the summary report.

The tool developed here requires only MS Access and MS Excel, both of which come with the MS Office Suite. Thus, it can be loaded easily onto a laptop and taken to the field. This will allow the program to be used by the crew members working on-site to
record immediate changes or by the officials in the office as well. The tool is menu
driven and is easy to understand and use.

The next chapter provides an overview of Visual Basic, Microsoft Access and
Microsoft Excel and discusses some of their key features.
CHAPTER 4

ABOUT VISUAL BASIC, MS ACCESS & MS EXCEL

Visual Basic, MS Access and MS Excel are "off-the-shelf" software programs used in the development of the MMS in this thesis. Key features of these programs are presented in this chapter.

4.1 Visual Basic

The following subsections give a brief overview of Visual Basic (VB). These include an introduction, basic concepts, the elements of a VB project, the components that make up VB and a functionality provided in VB that allows the program to be packaged and implemented on different computers.

4.1.1 Introduction

Visual Basic is a programming language developed by Microsoft Corporation. Visual Basic provides an extensive set of tools to simplify rapid application development. The "Visual" part refers to the method used to create the graphical user interface (GUI). Rather than writing numerous lines of code to describe the appearance and location of interface elements, pre-built objects can be added to the screen.
The "Basic" part refers to the BASIC (Beginners All-Purpose Symbolic Instruction Code) language. Visual Basic has evolved from the original BASIC language and now contains several hundred statements, functions, and keywords, many of which relate directly to the Windows GUI. Beginners can create useful applications by learning just a few of the keywords, yet the power of the language allows professionals to accomplish anything that can be accomplished using any other Windows programming language.

The Visual Basic programming language is not unique to Visual Basic. The Visual Basic programming system, Applications Edition included in Microsoft Excel, Microsoft Access, and many other Windows applications uses the same language. The Visual Basic Scripting Edition (VBScript) is a widely used scripting language and a subset of the Visual Basic language.

The following tools in Visual Basic make it very useful.

- Data access features allow the creation of databases, front-end applications, and scalable server-side components for most popular database formats, including Microsoft SQL Server and other enterprise-level databases.
- ActiveX™ technologies provide the means to use the functionality provided by other applications, such as Microsoft Word, Microsoft Excel, and other Windows applications. These applications can also be automated and objects created using the Professional or Enterprise editions of Visual Basic.
• Internet capabilities make it easy to provide access to documents and applications across the Internet or Intranet from within the application, or to create Internet server applications.

• The final application can be made into an executable file that can be packaged along with the dependent files to create a portable package that can run on an IBM compatible computer with MS-Windows NT/98/95 as the operating system. At least 32Mb of memory is recommended for good performance. Please read the specific hardware and software requirements in section 6.1 for further details.

4.1.2 Visual Basic Concepts

Visual Basic is a Windows development language that follows an Interactive Development Process. The window events are handled by Visual Basic and the programmer is isolated from the basic window management operations like refreshing a window or minimizing a window etc. Interactive development allows the user to test the application as it is being developed and this process is described below.

Interactive Development

The traditional application development process can be broken into three distinct steps: writing, compiling, and testing code. Unlike traditional languages, Visual Basic uses an interactive approach to development, blurring the distinction between the three steps.

With most languages, an error in writing code is caught by the compiler when the compilation of the application begins. The error must then be located and fixed, and the
program needs to be recompiled. However, Visual Basic interprets the code as it is entered, catching and highlighting most syntax or spelling errors on the fly.

In addition to identifying errors on the fly, Visual Basic also partially compiles the code as it is entered. If the compiler finds an error, it is highlighted in the code. The error can be fixed and the compiling continued without having to start over.

4.1.3 Elements of a Visual Basic Project

The following sections describe the different types of files and objects that have been included in developing the MMS in this thesis. The Visual Basic documentation [29] provides more information about these components or other components not mentioned here.

4.1.3.1 Files and Objects:

A VB project consists of user interface elements called forms. These are what are visible to the user for interacting with the program. These forms are stored in files called form modules that VB interprets and displays on screen. Each form has associated program code that determines the behavior of the form and also many standard modules that contain variable declarations.

Form Modules:

Form modules (.frm extension) contain textual descriptions of the form and its controls, including their property settings. They can also contain form-level declarations of constants, variables, and external procedures; event procedures; and general procedures.
Standard Modules:

Standard modules (.bas extension) contain public or module-level declarations of types, constants, variables, external procedures, and public procedures.

4.1.3.2 Components

The following components have been used in developing the MMS in this thesis.

ActiveX Controls:

ActiveX controls (.ocx file name extension) are optional controls that can be added to the toolbox and used on forms. One such control is the ActiveX Data Object Data Control (ADODC). This control is described next.

The ADODC uses Microsoft ActiveX Data Objects (ADO) to quickly create connections between data-bound controls and data providers. Data-bound controls are any controls that feature a “Data Source” property.

Although the ActiveX Data Objects can be used directly in applications, the ADODC has the advantage of being a graphic control (with Back and Forward buttons as shown in Figure 4.1) and an easy-to-use interface that allows the creation of database applications with a minimum of code writing.

Figure 4.1 The ActiveX Data Object Data Control
The ADODC can be easily used for the following purposes and is hence very useful:

- Connect to a local or remote database.
- Open a specified database table or define a set of records based on a Structured Query Language (SQL) query or stored procedure or view of the tables in that database.
- Pass data field values to data-bound controls, where the values can be displayed or changed.
- Add new records or update a database based on any changes made to the data displayed in the bound controls.

Many controls can be bound to the ADODC control and this provides a convenient way to view the database and select items based on the fields in the database. The bound controls used in this application are described below.

**Option Button:**

The option button control is shown in Figure 4.2. The option button control allows the user to select a particular option. The importance of the option button control is that it allows the user to select just one option from a given set.

![Option button Control](image)

*Figure 4.2 Option button Control*

**Check Box:**

The check box control is shown in Figure 4.3. The check box control displays a check mark when it is selected. It is commonly used to present a Yes/No or True/False
selection to the user. The check box control is similar to the option button control in that each is used to indicate a selection that is made by the user. They differ in that only one option button in a group can be selected at a time. With the check box control, however, any number of check boxes may be selected.

![Figure 4.3 Check box Control](image)

**Figure 4.3 Check box Control**

**Combo Box:**

A combo box control (shown in Figure 4.3) combines the features of a text box and a list box. This control allows the user to select an item either by typing text into the combo box, or by selecting it from the list. Combo boxes present a list of choices to the user. If the number of items exceeds what can be displayed in the combo box, scroll bars will automatically appear on the control. The user can then scroll up and down, or left to right through the list.

![Figure 4.4 Combo Box](image)

**Figure 4.4 Combo Box**

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
DBGrid Control:

The DBGrid control is a spreadsheet-like bound control that displays a series of rows and columns representing records and fields from a Recordset object. The DBGrid allows the end user to read and write to most databases. The DBGrid control can be quickly configured at design time with little or no code. When the DBGrid control's DataSource property is set at design time, the control is automatically filled and its column headers are automatically set from the data source's recordset. The grid's columns can then be deleted, rearranged, column headers can be added, etc.

At run time, the DataSource can be programmatically switched to view a different table, or the query of the current database can be modified to return a different set of records. The DBGrid can be bound to the ADODC that is connected to a database and all changes to the DataSource or the query is passed to the ADODC and the change is reflected in the DBGrid.

Standard Controls:

Standard controls are supplied by Visual Basic. Standard controls, such as the command button or frame control, are always included in the toolbox, unlike ActiveX controls and insertable objects, which can be removed from or added to the toolbox.

4.1.4 Converting the Visual Basic project to an Executable

The usefulness of Visual Basic is further extended by the facility to convert the project into an independent package. This package can be made to include all the necessary files including VB library and ActiveX files that are required to run the...
application. This is made possible with the help of one of the Add-in Wizards as described in the following section.

4.1.5 Package and Deployment Wizard

The Package and Deployment Wizard is a tool that helps create an installation package for the Visual Basic application and install it to end-users' computers. The Package and Deployment Wizard provides a number of options for distributing the application, including the creation of:

- standard packages designed to be installed by a "setup.exe" program;
- internet packages designed to be downloaded from a Web site; and
- dependency files that can be distributed with the components.

The wizard packages the application components into compressed cabinet (.cab) files. These files contain all the library files that are needed to run the application. The standard files referenced by the project file are automatically included in the package. However, if there are some user created files, e.g., a "ReadMe" file, that need to be included, those need to be added using one of the menu options of the wizard. Single "cab", or multiple "cab" files can be created that can be copied onto floppy disks. The wizard also creates a "setup.exe" program that will install all the components needed to run the application.

4.2 Microsoft Access

Microsoft Access (MS Access) is part of the Microsoft Office Suite of Products for Office and Business Automation. Microsoft Access 97 for the Windows® 95 and Windows NT® operating systems provides relational database power to make the
information available when it is needed to help make better decisions. It integrates data from spreadsheets and other databases and is the easy way to find answers, share information over Intranets and the Internet, and build faster business solutions [30].

Microsoft Access offers greatly enhanced 32-bit performance, including smaller forms, more efficient compilation, and better data manipulation technology. Other improvements substantially reduce the technical know-how needed to build fast, flexible business solutions. The Performance Analyzer Wizard automatically recommends the best way to create a well-organized, more responsive database. And the intuitive, integrated development environment (IDE) in Microsoft Visual Basic® for Applications (VBA) with Microsoft ActiveX® support makes it easy to develop a powerful database and integrate it with other Microsoft Office programs to create a comprehensive business solution [16].

MS Access databases can be converted to Hypertext Markup Language (html) pages that can be published on the Internet. Hyperlinks can also be embedded which provides a quick link between forms, reports, and tables within Microsoft Access databases, or other Microsoft Office documents and internal or external Web sites. MS Access is compatible with other database structures and allows importing or exporting databases from other formats.

An MS Access database can have a maximum size of 1 gigabyte including all the modules such as tables, forms, reports, queries etc. However there may be links to tables in other files and databases and hence the size is only limited by the available resources.

In this project, MS Access is used only to store the data in different tables. All the operations such as querying, report generation, analysis, updating etc., are carried out by
the Visual Basic program. The program accesses the database externally and is not built as a VBA interface. This helps in future development so that any external database can be connected to by just changing the driver.

4.3 Microsoft Excel

Microsoft Excel (MS Excel) is a spreadsheet program and is part of the Microsoft Office Suite of products. Microsoft Excel 97, for the Windows® 95 and Windows NT® Workstation operating systems, makes it easier to analyze, report, and share data [31].

A spreadsheet is a representation of data in the form of rows and columns much like a table. However in a spreadsheet each cell is represented individually. A cell is referred to by the number of the row and the column that intersect at that cell. For example, a cell formed at the intersection of Row 10 and Column B is referred to as B10.

A cell in a spreadsheet typically stores a certain value or a formula to calculate a value. The value may be a number, text, symbol etc. Each cell in a spreadsheet is an independent entity and can be programmed to perform independently of the other cells or on the other hand be used to represent values dependent on values in other cells or cells in other spreadsheets.

MS Excel is updated immediately. This means that if a cell is given a new value, all the cells that depend on that value get updated to represent the change. Complicated formulae and conditional statements (if....then...) can be used to program MS Excel to carry out analysis.

In this thesis MS Excel is used to generate a report. A template of the report has already been prepared and is part of the application. When the user chooses a certain
project to obtain a summary report, the application retrieves the required values from the MS Access tables and populates the cells in the Excel sheet report template. The template is then shown to the user by starting an MS Excel application. The report is formatted such that the user can take a printout as it is.

The reason for using MS Excel was that the individual cells can be populated with values independent of the others. Also it is a simple software to use and is available with the MS Office Suite. This keeps the cost of the complete application low and also allows users to change and format the report generated.

The next chapter describes in detail, the design of the roadway maintenance management system developed in this thesis.
CHAPTER 5

DESIGN OF THE ROADWAY MAINTENANCE
MANAGEMENT SYSTEM

The Roadway Maintenance Management System developed in this thesis has been
designed to be simple and yet useful. The system provides the basic functionality and can
be used as a useful decision support tool. The following sections describe the design of
the system.

5.1 Brief Overview

The roadway MMS developed in this thesis is built on a database stored in MS-
Access along with a VB user interface. The entire user interface has been developed in
VB and the user does not have to work with the MS access database directly.

The database is divided into several tables. Each of the tables stores information
concerning one particular type of resource. For example, one table stores only personnel
information while another stores only material information. Each of the tables however
has a common field that helps link the tables together and this is the project ID field.
Thus, all the correlation and querying is performed using the project ID field.
There are certain temporary tables also that are used while changes are being made to the main tables. The temporary tables allow the user to cancel the operations and undo the changes.

There is one person designated to be responsible for the security of the database and this person is referred to as the “Administrator”. The administrator can add and remove users and provide new users with passwords to access the database. The users can be selectively provided access to tables on an as needed basis and thus the information can be protected.

5.2 Basic Design

The basic design is divided into four parts. The flow of control section describes the overall structure of the application. This is depicted in the form of a flow chart in Figure 5.1. The “Application Design” subsection describes the main modules of the program and the “Database Design” subsection describes the tables that store the information. The “Interface Design” subsection describes how the interface has been designed.

5.2.1 Flow of Control

The flow of control describes the sequence of events and their dependencies. A flow chart shown in Figure 5.1 summarizes the structure of the application. All the different windows and forms are not shown. The flowchart only shows the generic design.
Figure 5.1 Flow of Control
Each command button (function) leads to a different branch of the program and there are a large number of combinations. It is very difficult to describe every permutation here, however the basic flow of control is described in the following paragraphs.

The application begins with a “Main Login Screen”. This prompts the user to enter a username and a password, provided by the Administrator. The user can change the individual password but cannot change the username.

The username and password are verified with that stored in the database and if successful, the “Welcome” screen appears. This screen stays for a few seconds while the program is loaded. If the login is unsuccessful then the user is given two more chances (three chances in all) to login correctly. If the user does not succeed all three times then the program shuts down and needs to be started again.

The main screen called “Task Manager” (TMgr) appears after the welcome screen. It is from the Task Manager screen that all the functions can be used. The TMgr is the fulcrum of the application and all functions start and end with this screen. The administrator can permit or deny each user the permission to access the tables by changing the “User Permissions”. Only the administrator can change the permissions.

The following section describes, in detail, the design of the application.

5.2.2 Application Design

The entire application is centered around working with information stored in a database. In this case the information is stored in Access tables. Several people may use the system and hence the database needs to be password protected.
One of the controls available in VB that can be used to connect to password protected databases is the ActiveX Data Object Data Control (ADODC). This also supports Sequential Query Language (SQL) statements as search criteria while retrieving records. SQL is a simple query language that is used to search and manipulate records in a database. ADODC supports all SQL statements. ADODC also allows the records to be deleted, added, updated, edited etc. This flexibility of ADODC was the primary criterion that helped in choosing ADODC as the control to be used for connecting to the database.

Once a connection is established with the database using the ADODC, the data should be made accessible to the user. Many controls are available that can be used to display and manipulate the data. These controls are called bound controls. This means that these controls are bound to the ADODC and hence the database. Thus, any change made to the records through these controls can be made to reflect in the database using the ADODC. Correspondingly any change in the database is immediately reflected in the control. One such control is the Data Bound Grid control (DBGrid). This is used to display tabular data and can be bound to an ADODC. The DBGrid also has several functions that will be described later that were found to be very useful in displaying the data.

The basic design thus involves ADODC and DBGrid. All the data retrieval operations and manipulations are handled by the ADODC and are displayed using a DBGrid.

The efficiency and ease of use of this system is dependent on the database and what each table contains and how the information can be managed. The database is made
up of tables with one common item that relates all the tables together as described in the next section.

5.2.3 Database Design

The database used is an MS Access Database. This was chosen for its simplicity and low cost. Also, the application is designed for use by small offices that would normally have MS Access as part of the MS Office suite. The database also had the necessary requirements such as ability to store tables and password protection.

There are two databases that are used to run the application. One of them called "admin.mdb" has the passwords for the administrator, the path to the main database, and the password for the main database. This database is also password protected and the password for this is coded in the program. Hence this password cannot be changed but there is no need to do so either. The second database called "main.mdb" is the database that has all the information and stores all the data for the entire maintenance management system. The "main.mdb" can be renamed and relocated and the changes are recorded in the admin.mdb database. The "admin.mdb" file needs to be in the same location as the application executable file.

The database consists of data stored in the form of tables. All the tables are described later in this section. The tables for this application are of two types called Basic and Project. The Basic tables store information about the resources available. For example, the details about the crew such as the name, wage rate ($/hr), designation, etc are stored in a table called “cp1”. The basic tables appear with descriptive names in the
TMgr screen. There are similar tables for equipment, material, street details and maintenance activity.

The second type of tables is called “Project” tables. These are tables that store information about the different projects or work orders. Similar to the basic tables there is a main project table along with tables for crew/personnel, equipment, material and expenses. These tables access the basic information like street segment number, crew name etc from the “Basic” tables mentioned above. Hence the information should be entered into the basic tables before any information regarding projects can be stored.

Apart from these fundamental sets of tables, there are temporary tables that are used to temporarily store some information while the user is changing the records in one of the tables. This allows the changes to be reversed and the original data can be restored if the user makes a mistake. However once the user saves the changes, the original cannot be recovered.

There are two other tables, one that stores the passwords and permission details for each user, and another that stores the name of the organization and the department using this application. These names appear above the summary report.

The expenses connected with each project are calculated and stored in a table within the database. This table stores the total cost of crew, equipment, material, the total cost of the project, and the total amount already paid. The values in these tables are calculated from the other project tables. This table cannot be changed or appended to, through the TMgr. This table is updated internally when the other concerned project tables are changed. For example, if the amount paid to a member of the crew is changed this table gets automatically updated. However this function can be viewed and queried.
The project tables store information for each project. The main project table has one line for each project with information such as the “Project ID”, the “Work Order Number”, “On-Street” etc. The crew project table has one line for each member of the crew working on a particular project. Similarly there is one line for each material or equipment entry. This application has a limit of a maximum of 4 entries at a time. Any additional entries need to be made by running the input function one more time. It is to facilitate this that there are two fields connected to each project the Project ID and the Work Order Number. The additional (more than 4) crew, equipment, material can be entered with a different work order number. However, the general practice is to have all the details under the same Project ID and Work Order Number. Hence the information can be entered with the same details because this application does not perform any checks to make sure that the Project ID or Work Order Number does not already exist.

In order to save time spent on corrections, the user needs to make sure that the information is correct. However the information entered can be changed at any time. The expense table is the only table that is updated dynamically each time one of the connected tables is changed. For example, the addition of a member to the crew will cause the total cost fields in the expense table to be updated.

The aforementioned tables are described next with specifications and a brief description about each field:

BASIC Tables: The basic tables store the information that is used by the application to provide choices to the user for entering information. The basic tables are also used to
calculate the cost of using each resource. All the fields in basic tables need to be entered by the user before beginning to use any of the functions.

A)  

Table Name: "cp1"

Purpose: Store Information about the Personnel such as name, designation etc.

Fields available:

1) Record ID: Internal record identifier. Integer. No default value.

2) Name: Name of the individual. The name can be entered in any form as long as the same format is followed throughout the application. Text. Maximum length 200. No default value.

3) Designation: Designation such as “Supervisor”, “Staff”, “Driver” etc. There is no fixed list of designations and hence new ones can be added at any time. Text. Maximum length 200. No default value.

4) ID: Identification number or keyword. This may be used to identify a particular crew member instead of the using the complete name. Text. Maximum length 200. No default value.

5) Team: The name of the team or group, if any, to which the crew member belongs. Text. Maximum length 200. No default value.

6) Wage ($/hr): The rate at which the crew member is to be paid. Number. Currency type format. No default value.
B) Table Name: “mat2”

Purpose: Store Information about the Material such as name, unit of measurement, unit cost, etc.

Fields available:

1) Record ID: Internal record identifier. Integer. No default value.

2) Name: Name of the material such as concrete, aggregate etc. Text. Maximum length 200. No default value.

3) Unit (lb., tons, etc): Unit of Measurement such as pounds (lb), gallons (gal) etc. Text. Maximum length 50. No default value.

4) Unit Cost ($/unit): Cost per unit. Currency type format. No default value.

C) Table Name: “eq3”

Purpose: Store Information about the Equipment such as name, unit of measurement, unit cost, etc.

Fields available:

1) Record ID: Internal record identifier. Integer. No default value.

2) Name: Name of the equipment. Text. Maximum length 200. No default value.

3) Unit (hrs, miles, etc): Unit of measure of how the equipment cost is calculated. For example the cost of a truck may be in hours whereas the cost of a road paver may be measured in units of miles of paving done. Text. Maximum length 50. No default value.

4) Unit Cost ($/unit): Cost per Unit. Currency type format. No default value.
D) Table Name: “str5”

Purpose: Store Information about each segment of streets such as the name of the street, the length of the segment, the cross-street names etc.

Fields available:

1) Record ID: Internal record identifier. Integer. No default value.

2) Name: Name of the main street. Text. Maximum length 200. No default value.


4) From: Name of the first cross street, designated as the “from street”. Text. Maximum length 200. No default value.

5) To: Name of the second cross street designated as the “to street”. Text. Maximum length 200. No default value.

6) Total # of Lanes: Number of Lanes on the segment in both directions. Integer. No default value.

7) Width (feet): Average width of a lane on the segment. Number with 2 decimal places. No default value.

8) Pavement Type: The type of pavement such as rigid, flexible. Text. Maximum length 200. No default value.

9) Surface Type: Surface type such as asphalt concrete. Text. Maximum length 200. No default value.

10) Length (miles): The length of segment under maintenance. Number with 2 decimal places. Default value is the Length of the Segment.
E) Table Name: “passwd7”

Purpose: Store Information about the users such as their login name, password etc.

Fields available:

1) Record ID: Internal record identifier. Integer. No default value.
2) Name: The login name of the user. Text. Maximum length 200. No default value.
3) ID: Identification number or keyword. Need not be same as that in the personnel table (“cp1”). Text. Maximum length 200. No default value.
5) Group: The name of the team or group, if any, to which the crew member belongs. Need not be same as that in “cp1”. Text. Maximum length 200. No default value
6) to 15) tbl to tbl0: Each stands for a particular table. These have values of either 1 or 0 depending on whether the user is given permission to access the table or not.

F) Table Name: “maintact8”

Purpose: Store Information about each Maintenance Activity Data such as the name and identifier number or keyword.

Fields available:

1) Record ID: Internal record identifier. Integer. No default value.
2) ID: Identification number or keyword that identifies the maintenance activity. Text. Maximum length 200. No default value.

3) Description: Name of the Activity. Text. Maximum length 200. No default value.

G) Table Name: “info”

Purpose: Store Information about the Organization using the application such as the name of the department. These are the two items that will appear as headings on the summary reports.

Fields available:


PROJECT Tables: These tables are derived from the basic tables and from user input. A record is added for each new project that is added. These tables are initially empty and get data when the details about the project are entered. Most of the values in these tables can be chosen from drop down menus or can be typed in anew.

A) Table Name: “main4”

Purpose: Store details about each project such as the street and segment name that is being affected etc.

Fields available:
1) Record ID: Internal record identifier. Integer. No default value.


3) Work Order Number: An alphanumeric string that identifies the work order number. Text. Maximum length 200. No default value

4) On-Street: Name of the Street on which the Activity is being carried out. Text. Maximum length 200. No default value

5) Segment: Segment Name. Text. Maximum length 200. No default value

6) Activity Miles: Length in Miles over which the Activity is taking place. Number with 2 decimal places.

7) Affected Lanes: Lanes on which the Work is being done. Number.


9) Foreman: Name of the foreman in any format as long as the format is consistent with the basic personnel table ("cp1"). Text. Maximum length 200.

10) Starting Date: Date.(mm/dd/yyyy)

11) Ending Date: Date.(mm/dd/yyyy)

B) Table Name: "costs6"

Purpose: Store Information about Project Costs such as total personnel cost, total equipment cost etc

Fields available:

1) Record ID: Internal record identifier. Integer. No default value.
2) Work Order Number: An alphanumeric string that identifies the work order number. Text. Maximum length 200. No default value.


8) Total Cost: Total Cost of the Project. Number with 2 decimal places.


C) Table Name: “mainpers”

Purpose: Store Information about the Personnel involved in Projects such as name, number of hours etc.

Fields available:

1) Record ID: Internal record identifier. Integer.

2) Name: Name of the crew member in the format that is consistent with the basic personnel table. Text. Maximum length 200.

3) Number of Hours: Number of hours for which the crew member should be paid. Number with 2 decimal places

5) Total Wage: The total wage that the crew member earned for this project. This value is calculated automatically from the number of hours and the wage rate. Number with 2 decimal places.

6) Paid: Amount already paid to the crew member. Number with 2 decimal places.

D) Table Name: “maineq”

Purpose: Store information about the Equipment assigned to a Project such as the name, the number of units etc.

Fields available:

1) Record ID: Internal record identifier. Integer.

2) Name: Name of the equipment. Text. Maximum length 200.

3) Number of Units: Number of units of equipment used for the project. Number with 2 decimal places


5) Total Cost: Total cost for the equipment. This value is calculated from the number of units and the cost per unit. Number with 2 decimal places

6) Paid: Amount already paid for the equipment. Number with 2 decimal places

E) Table Name: “mainmat”

Purpose: Store information about the Material assigned to a Project such as the name, the number of units used etc.
Fields available:

1) Record ID: Internal record identifier. Integer.

2) Name: Name of the material. Text. Maximum length 200.

3) Number of Units: Number of units used for the project. Number with 2 decimal places


5) Total Cost: Total cost for the material. This value is calculated from the number of units used and the cost per unit. Number with 2 decimal places

6) Paid: Amount already paid for the material. Number with 2 decimal places

There are temporary tables for each of the tables above. The temporary tables are used while changing or editing the main tables. This is so that the user has the opportunity to cancel a change operation. However the old data cannot be recovered once the changes are saved.

The following section describes the design of the interface. The interface is meant to interact with the user and facilitate easy input and use of the application. Hence the design of the interface and the controls therein are very important.

5.2.4 Interface Design

The interface has been entirely built using Visual Basic 6.0 (VB). VB was used since it an easy to use language and yet robust enough to provide all the functionality needed. The basic functionality needed was a method to connect to and manipulate secure databases. Combination boxes, command buttons and grid structures (to display tables)
were some of the other needed features. Since these features were provided by VB and since VB is a simple language to program in, VB was chosen for developing the user interface.

The interface mainly consist of simple command buttons and easy to use choices. The command buttons can be operated using the “Hot Keys”. The hot key is activated by holding down the “ALT” key along with the letter underlined in the command button. Also to make it easier for the user, the default button is set as the button that may most probably be chosen. For example, pressing the “Enter” key will lead to the default command being executed.

By Default all the windows appear at the center of the screen. All the windows can be minimized, maximized or closed. Closing the window is not the same as pressing the cancel button and may cause unpredictable consequences. It is strongly recommended to press the “Cancel/Return” button to close a window. Usually closing a window using the “X” sign at the top right corner of the window causes the window to close but the program continues to run in the background waiting for the “cancel” event of the window to occur. Thus, the database will get locked and cannot be accessed by any other program. The program running in the background will have to be ended using the “Windows Task Manager”. The window can however be minimized or maximized. At any time there will be only one window that is visible. The window can be moved around the screen.

In order to make the input of data easy, wherever possible input textboxes have been replaced by “Combination Boxes” (Combo Boxes). These boxes have a drop-down
list that shows the choices available from which the user can either choose one or type in a new value.

Every effort has been taken to make the forms appear as simple and concise as possible, but in some cases it was necessary to display a lot of information in one screen.

The following section describes the functions that are available to manipulate and manage the database. These functions are used to input and retrieve data and perform queries.

5.3 Available Functions

There are several functions available to the user for data entry and manipulation. The functions allow the user to maintain the database that stores all the information. All the functions have been developed by writing VB code and none of them are built-in functions of VB or Access. The functions available are the following.

a) View
b) Append/Add
c) Change
d) Analyze/Query
e) Report
f) Change Password

Each of these functions is available for most of the tables. However some restrictions apply. For example, it is not possible to change the expense table. This is because it is a "derived" table with data calculated from other tables. As mentioned earlier, access to the tables may be restricted to users by the administrator. However if the user is given
permission to access a table, then that user can utilize all the functions allowed for that table. In other words, it is not possible to restrict the functions for a particular table but only the entire table. Thus, the user gets to use all the functions for a table or none at all. These functions are explained in the following sections.

5.3.1 View Function

Objective:

This function allows the user to view a table.

Limitation:

The user is only allowed to view the information and cannot make any changes.

Design:

This has been developed using the DBGrid control bound to the ADODC. This control provides a dynamic column-row (grid) structure that can be connected (bound) to a data source (ADODC). The control does not have a restriction on the number of columns or rows and thus a table with any number of rows and columns can be displayed. The control takes care of allocating memory and displaying the required number of fields and records. The records are displayed in ascending order of the Record ID.

Using the Function:

The user can select to view any of the tables that are enabled for that user. Once the selection is made the "View" button is clicked (Hot key: "ALT" + "V"). Another window displaying the table in a DBGrid comes up. Scroll bars are automatically attached in case the columns or rows go out of boundary of the view. The user can scroll and view all the elements in the table but cannot make any changes. The "Cancel/Return"
button needs to be clicked (Hot Key: "ALT" + "C") to return to the previous window (TMgr). A snapshot of the View window for the crew/personnel table is shown in Figure 5.2:

![Table View]

**Table View**

<table>
<thead>
<tr>
<th>Record ID</th>
<th>Name</th>
<th>Designation</th>
<th>ID</th>
<th>Team</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ackiss, John</td>
<td>Worker</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Willmott, Kim</td>
<td>Worker</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Birt, David</td>
<td>Assistant Supervisor</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>Minor, Andrew</td>
<td>Staff</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Woolman, Craig</td>
<td>Worker</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Chavez, Jimmy L</td>
<td>Staff</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Allison, William</td>
<td>Worker</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>Berry, Michael</td>
<td>Supervisor</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>Vohs, Michael</td>
<td>Supervisor</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>Ouellette, Dave</td>
<td>Driver</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

Figure 5.2 View of the Basic Crew/Personnel Table

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
5.3.2 Append/Add Function

Objective:

To allow the user to add new records to the database

Limitations:

This function works on all the “Basic” tables and on the “Work Order (Main)” table. The records can only be added and therefore the older records cannot be changed.

Design:

The addition function has also been built with combination boxes and textboxes. The combo boxes are used where some information may be same as that already present in the database. For example the new personnel being added may have the same designation as one already present. However, a new designation can also be typed in by the user. An internal Record ID is assigned to the addition automatically. The user need not be concerned with this field. The combo boxes are also connected to the database via the ADODC.

Using the Function:

The user selects a table from the TMgr screen and clicks “Append/Add” button (Hot Key: “ALT” + “A”). This brings up a new form depending on the table that was chosen. The user can enter all the values by choosing from the drop-down combo boxes or typing values into the textboxes wherever appropriate.

Once all the values are entered the user clicks “Save” (Hot Key: “ALT” + “S”). This saves the new record. If the user wishes another record can be added by clicking “Add Another Personnel” (in this example, “Personnel”) (Hot Key: “ALT” + “A”). The user can otherwise choose to return to the previous menu by clicking the “Cancel” (Hot
Key: “ALT” + “C”) button. Anytime the “Cancel” button is clicked any changes that were not saved will be lost.

All the fields are usually necessary and before any saves are affected, the program checks for the completeness of the information. If there is something missing, that is required, then the user is prompted with an error message. A snapshot of the Append window for adding crew/personnel is shown in Figure 5.3.

![Personnel Information Input Form](image)

Figure 5.3 Append/Input Window for Crew/Personnel
5.3.3 Change/Edit Function

Objective:

To allow change (edit) the records in the table

Limitations:

The record ID cannot be changed for any of the tables. Only the “Paid” field can be changed in the “Crew/Personnel”, “Material Used” and “Equipment Used” amongst the “Project” tables. Other fields in these tables can only be changed by changing the “Work Order (Main)” table.

Design:

The change/Edit function has been designed differently for the “Basic” tables and the “Project” tables. The “Basic” tables are changed using a DBGrid and the ADODC like the “View” function. This was done so that the user can scroll down to any record and make changes. The first column displays the internal “Record ID”. This is locked at run time, when the form is being loaded, to prevent the user from changing this field.

For the ”Project” tables (except the main table), however, only the “Paid” field can be changed. All other fields are locked. This is done because the amount paid to that entity may not be known in the beginning and hence it may be necessary to enter/change the value later.

The other fields related to each project like the number of hours of a member of the crew need to be changed by changing the “Work Order (Main)” table. This brings up a window similar to the “Add” function for the main project table. Any of the fields except the “Record ID” can be changed.
Using the Function:

The user chooses a table and clicks “Change/Edit” button (Hot Key: “ALT” + “C”).

For the “Basic” tables a grid like view will appear (like the “View” screen) and the fields can be changed. More than one field and record can be changed and at the end the “Save” can be clicked (Hot Key: “ALT” + “S”).

For the “Project” tables except the “Work Order (Main)” table, only the “Paid” field can be changed. This also appears as a grid.

For the “Work Order (Main)” table a form similar to the “Add” window will appear. Only the “Project ID” field will be enabled at this point. The user needs to choose a Project ID from the drop down combo box. As soon as this is done the other fields get populated with the values from the database. The fields can then be changed. The user then needs to click “Proceed/Continue” (Hot Key: “ALT” + “P”) in order to change other details like the crew, material or equipment. Once the changes are made the “Save” button should be clicked to save the changes.

A snapshot of the change menu for “Work Order (Main)” table is shown in Figure 5.5. The second screen as shown in Figure 5.5, appears when the “Proceed/Continue” button is clicked. The changes made on the first screen can be saved only by proceeding to the second screen and clicking on “Save”. Figure 5.4 shows the first screen. All except the “Project ID” field are disabled.
Figure 5.4 Change/Edit Window for the Main Project Table
### Personnel/Material/Equipment Entry Form

<table>
<thead>
<tr>
<th>Name of the Personnel</th>
<th>Number of Hours</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Crew Member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Crew Member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Crew Member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Crew Member</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Equipment

<table>
<thead>
<tr>
<th>Equipment Name</th>
<th>Equipment Units</th>
<th>Number of Units</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Material

<table>
<thead>
<tr>
<th>Material Name</th>
<th>Material Units</th>
<th>Number of Units</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5.5 Personnel/Material/Equipment Input Window
5.3.4 Analyze/Query Function

**Objective:**

To analyze and search for records based on certain criteria.

**Limitations:**

This function is based on a menu driven query builder and hence only those options that are provided can be used and a general SQL statement cannot be given.

**Design:**

The analysis and query is based on building a query statement at run-time based on the options chosen by the user. This query statement is built as a SQL statement that is passed to the ADODC as the selection criteria. The records that are returned are displayed in a DBGrid.

There are a large number of queries that the user may have depending on the table. It is a very difficult task to develop a completely robust and error free query builder and hence the options were limited. A different query builder has been developed for each table with an aim to cater to most of the users' queries. The query allows the user to choose fields from drop-down menus for the fields that contain text values such as “Project ID” or “Name” fields. For the fields that have numerical values, there are options to get the maximum, average, minimum, or greater than (>) or less than (<), or equal to (=) a certain value. The user is cautioned to be careful while using the query builder. Though every effort has been taken to make it as robust as possible and detect errors, some permutations may have been missed out. An error might result in the application being terminated.
All queries are based on the “AND” operator. Hence if a “Project ID” and a “Name” is chosen then the resulting records will be those that contain that project id AND that name.

A snapshot of the Analyze/Query screen for the Personnel table is shown in Figure 5.6.

![Personnel Query Builder](image)

**Figure 5.6 Analyze/Query Window for the Personnel Table**

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Using the Function:

The user needs to choose the particular table and then click "Analyze/Query" (Hot Key: "ALT" + "Q"). A new form will appear, similar in basic structure to that shown above, depending on the table chosen. The combo boxes are bound to the ADODC and will contain values of records already present in the tables. The user may choose values from the drop-down menu or may select one of the check boxes for other options. Only one of the check boxes in each column can be chosen at any one time. The application checks for errors of this nature. For options that involve a form of equality, a numeric value should be entered in the textbox at the bottom.

Once the options are chosen then the user needs to click "Show" (Hot Key: "ALT" + "S"). The program will form a SQL statement with the options that were chosen by the user. The SQL statement is passed to the ADODC and records are selected based on this SQL query. The application does run some checks on the SQL statement to make sure that it makes sense. If the user has requested an analysis (like maximum value) then the calculated value is displayed in a message box and then the selected records are displayed. The selected records are shown in a DBGrid in another window.

5.3.5 Report Function

Objective:

To generate a summary report for a chosen project.

Limitations:

The format for the report is fixed.
Design:

The report is generated as an Excel sheet. The report has a fixed format and the summary is built on that fixed template. The application starts a new excel application and brings up the template sheet with all the summary values filled up. The Excel application is run independently of the roadway MMS application and hence the user is free to use the summary as any normal Excel sheet. The page setup is however done at run time while the summary is being generated and hence the user can instantly obtain a printout. The summary consists of the total cost of crew, equipment and material, the total cost of the project, the cost per mile and the total paid and the difference left to be paid.

The summary format is based on that used by the Maintenance Management Division of Clark County Department of Public Works (CCPW) (Appendix A). However, this report does not contain as many details as the CCPW format though this also gives most of the necessary information. A snapshot of the Excel “Summary” sheet template is shown in Figure 5.7. The titles for the report cannot be seen here because those are filled in during run-time. The titles will appear in the center at the top next to the date in the actual report.

Using the Function:

The user needs to click on the “Report” button (Hot key: “ALT” + “R”). This will bring up a list of project ids. The user needs to choose one of those listed and click “OK”. The application will start a new Excel application and bring up the Excel sheet report with all the fields filled in. The user can then use this as an independent excel sheet.
The user is cautioned to save the file with a new name. The template excel sheet is called "Temp.xls" and the application looks for this file in the home directory (the directory where the application executable exists) to create the report.

If some of the field values cannot be determined or are ambiguous (for example, the cost/lane-mile may not be defined for work at an intersection), then those field values are left blank. Being a simple Excel sheet, the user is free to fill in the missing values if known.

![Figure 5.7 Summary Report Format Template](image)

Figure 5.7 Summary Report Format Template

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
5.3.6 Change Password

Objective:

To change the user password.

Limitations:

There is no limitation though if the user forgets the password then the administrator will need to view the original password in the MS Access database to provide a new password.

Design:

The user can change the individual password to anything else and any number of times. The password along with the permissions is stored in the database and only the administrator has access to this information.

Using the Function:

The user needs to click on the “Change Password” button (Hot key: “ALT” + “P”). This will bring up a window asking for the old password and the space to enter the new password. The new password needs to be entered twice to confirm the change. This window is shown in Figure 5.8
The administrator can also change the administrator password and also the database password. The administration screen is shown in the next section.

5.4 Administrative Functions

The administrator is presented with an “System Administration” window showing the administrative functions as shown in Figure 5.9. The “System Administration” window will not appear to a non-administrative user. Once the application is started using the “Start Application” button, the administrator cannot return to “System Administration” screen without starting the application again.
System Administration

Change the Administrator's Password

Change the Database Password

Change/Add Users, Permissions etc

Change/View Official Information

Change the Database Location

(Current Location is C:\00\ranjit\project\main.mdb)

Start the Application

Exit

Figure 5.9 System Administration Window

The following chapter describes how the application can be installed and provides a sample project.
This chapter describes the steps to install and run the application. A sample tutorial project is also described.

6.1 Hardware and Software Requirements

The minimum hardware and software requirements needed to install and run the roadway MMS are: an IBM compatible computer with a Pentium type processor; at least 32Mb of memory and a VGA monitor; the operating system should be MS-Windows NT version 4.0 or Windows 95 or Windows 98; the MS-Office suite, specifically MS-Access and MS-Excel. The roadway MMS can be installed from a CD-ROM or a set of 3.5” floppy disks and hence a CD-ROM drive or a 3.5” floppy is needed for installation. The complete application along with all the necessary files occupies a minimum of approximately 13Mb of memory. However the total space required on the hard disk will increase with the size of the database. Also, a printer is needed to take printouts of summary reports generated by the application.
6.2 Installation

The installation is done in a simple manner by just running the setup.exe program. This will guide the user through the various installation steps and also copy all the necessary files. Once the installation is complete, an roadway MMS folder will be added to the Programs menu of the Start Toolbar. Alternatively the General.exe file may be run from the installation directory.

6.3 Using the application

The application requires one person to be designated as the “Administrator”. This person will be responsible for the database security and for maintaining the list of users allowed to use the system. The administrator has full control over the database and the user permissions. The administrator can add, delete or change permissions of users.

The administrator creates a new user account by creating a user name, creating a password and giving permission to access tables. This procedure is interface driven and can be done from the administrator’s choices. Once a new user account is given the user can change the password. However, the administrator will be able to view the new user password if it is required. Each user can only be denied or allowed access to an entire table. However, if the user is given permission to access a table, then all the functions associated with that table are available to that user.

6.4 Sample Application

The following is the description of a sample project that was run on the application. The sample describes how a new project is added to the database and how a
few sample queries are run. The sample shows snap-shots of the various screens that are used and each of the five functions available. On starting the application, the user is presented with the “Main Login Screen” as shown in Figure 6.1.

![Main Login Screen](image)

Figure 6.1 Main Login Screen

The user logs in with a username and password and a “Welcome” screen as shown in Figure 6.2 appears.

![Welcome Screen](image)

Figure 6.2 Welcome Screen

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
The welcome screen is a splash screen and will load the program within a few seconds and will be replaced with the "Task Manager" screen as shown in Figure 6.3.

The tables are enabled depending on the permissions set by the administrator allowing the user access to the certain tables. These permissions can only be changed by the administrator. However, if the user has access to a particular table then the user can carry out any of the Options available.

---

**Task Manager**

*Choose a table from the following*

**Basic Tables**

- Crew/Personnel
- Material
- Equipment
- Maintenance Activity
- Streets

**Project Tables**

- Work Order (Main)
- Crew/Personnel
- Materials Used
- Costs
- Equipment Used

**Options**

- View
- Append/Add
- Change/Edit
- Analyze/Query
- Reports
- Change Password
- Exit

---

*Figure 6.3 Task Manager Screen for a General User*
If the user logs in as an administrator then the "System Administration" screen, shown in Figure 6.4, will appear.

![System Administration Screen](image)

Figure 6.4 System Administration Screen

The administrator can start the application by clicking on the "Start Application" button that is also set as the default.

Note that the administrator can change password from the administration screen whereas any other user can change the password from the "Task Manager" screen.
Also, once the application is started the administrator cannot go back to the “System Administration” screen without starting over.

The following steps need to be followed for first time use of the database. The user needs the required permissions to work with any table.

1) For first use, the user needs to enter data in the “Basic” tables. These tables store the information about personnel, equipment, material, maintenance activity and the streets. Each of the tables is chosen by clicking on the option button and then the “Append/Add” button is clicked to add new data.

2) The data that was entered may be viewed using the “View” button and any mistakes can be corrected using the “Change/Edit” button.

3) Once the basic information is entered then the details about each project can be entered using the “Project Tables”. Choosing the “Work Order (Main)” table and clicking on the “Add/Append” brings on an input form with drop-down boxes and input boxes. The drop-down boxes will contain all the fields that are already in the database. The user may choose one of the items shown in the drop-down box or type in a new value. All the fields in this screen are required.

The segment number combo box gets values once an On-Street is chosen. Selecting a segment from the segment combo box will fill up the From and To street names and also the activity miles. However if the work is to be carried out in more than one segment then a new segment number may be entered or none at all and then the From and To street names can be typed in and also the activity miles.

A completed form is shown in Figure 6.5.
4) Once entries are checked for correctness, the user needs to press “Proceed/Continue” button. This will bring up the screen for inputting the personnel, equipment and material information.

The main drawback with this application is in this screen because it allows the user to enter a maximum of four values for each of the entities. Making the program...
more versatile by allowing an indefinite number of entries may be one of the avenues for future work. A completed form is shown in Figure 6.6.

![Personnel/Material/Equipment Input Form](image)

<table>
<thead>
<tr>
<th>Name of the Personnel *</th>
<th>Number of Hours *</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Berry, Michael</td>
<td>32</td>
<td>$960</td>
</tr>
<tr>
<td>Allison, William</td>
<td>40</td>
<td>$800</td>
</tr>
<tr>
<td>Woolman, Craig</td>
<td>25</td>
<td>$500</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Equipment Name *</th>
<th>Equipment Units</th>
<th>Number of Units *</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>92 Midland Cold Mix Paver 705</td>
<td>hrs</td>
<td>100</td>
<td>$12180</td>
</tr>
<tr>
<td>95 Chevrolet 3/4 Ton 4X4</td>
<td>hrs</td>
<td>50</td>
<td>$757</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Material Name *</th>
<th>Material Units</th>
<th>Number of Units</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cold Mix</td>
<td>Ton</td>
<td>100</td>
<td>$2200</td>
</tr>
<tr>
<td>Fill Material</td>
<td>Ton</td>
<td>100</td>
<td>$300</td>
</tr>
</tbody>
</table>

Figure 6.6 Completed Personnel/Equipment/Material Input Form

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
The input form for personnel/material/equipment also has drop-down menus to minimize typing and as the number of hours/units are entered the cost is calculated automatically using information from the "Basic Tables". Once all the entries have been made, the user should press the "Save" button. This saves the entire project information and takes the user back to the previous screen. This screen now displays the Total Cost for the project as shown in Figure 6.7.

![Main Input Form](image)

**Main Input Form**

*Record ID:*

*Project ID:*

*Work Order Number:*

*Starting Date: 11/15/99*  
*Ending Date: 11/10/99*  

*Maintenance Activity: General Paving*  
*On-Street: MARYLAND PKWY*  
*Segment: L145370E15*  
*From: FATE*  
*To: TWAIN*  
*Activity Miles: 0.13*  
*Affected Lanes: 2*

**Total Cost = $17697**

[Proceed/Continue]  
[Cancel/Return]

Figure 6.7 Final Main Input Form

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
5) The user can return to the “Task Manager” screen by clicking on “Cancel/Return”. Once at the main screen, the user can view the project that was just added. A Table view with scroll bars will show up with some of the essential information. The view function for the other entities like the personnel, equipment, material etc shows all the records. In order to view particular records, the “Analyze/Query” tool should be used. The table view of the project is shown in Figure 6.8 first, followed by the results of the query performed on personnel in Figure 6.9.

![Table View](image)

**Table View**

**Work Order (Main)**

<table>
<thead>
<tr>
<th>Project ID</th>
<th>Work Order Number</th>
<th>On-Street</th>
<th>Segment</th>
<th>Act</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB345</td>
<td>CD678</td>
<td>MARYLAND PKWY</td>
<td>L145370E15</td>
<td>0.13</td>
</tr>
</tbody>
</table>

Figure 6.8 Table View of the Project

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
### Analysis Results

#### Crew/Personnel

<table>
<thead>
<tr>
<th>Name</th>
<th>Number of Hours</th>
<th>Project ID</th>
<th>Total Wage</th>
<th>Paid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Berry, Michael</td>
<td>32</td>
<td>AB345</td>
<td>960</td>
<td>0</td>
</tr>
<tr>
<td>Allison, William</td>
<td>40</td>
<td>AB345</td>
<td>800</td>
<td>0</td>
</tr>
<tr>
<td>Woolman, Craig</td>
<td>25</td>
<td>AB345</td>
<td>500</td>
<td>0</td>
</tr>
</tbody>
</table>

![Figure 6.9 Results of a Query on the Personnel Table](image)

6) The "Paid" field shown here is zero and can be changed by selecting the personnel project table and using the "Change/Edit" function from the main "Task Manager" screen. This will update the cost table as well.

7) A summary report can be prepared for any particular project that is chosen from the "Project Names" screen shown in Figure 6.10. The report format is fixed, however, it has been designed to show most of the common details that are needed. The report function is activated from the Task Manager screen and this brings up the list of
projects available. The user can choose a particular project for which to prepare the report.

![Project Name Selection Window](image)

**Figure 6.10 Project Name Selection Window**

8) The report is prepared in Excel. An Excel application will be started and the summary report is shown. This report is set to a portrait 8.5 inches by 11 inches page size with 0.75 inches of border margin on all sides. The report is thus set for printing and can be printed as it is. If the user wants to save the report then it has to be done with a new name. The report comes up with the name "tmprpt.xls" and this file is overwritten each time a summary report is prepared and hence the file should be saved under a new name.
The screen-shot of the Excel sheet report is shown in Figure 6.11. The Excel sheet is maximized to fill up the screen and the Task Manager window behind is not visible. However before using other functions from the Task Manager, it is recommended that the Excel sheet be closed.

Figure 6.11 Summary Report

This concludes the sample showing the key features and functions of the roadway MMS tool. There are many other features and analysis and query options that have not been described here. These may be tried out with the sample data provided with this application.
6.5 Useful Features of the Application

As mentioned earlier, the application can be used to prepare a summary report of any particular project. This summary report provides the supervisor using the system with an idea of the total expenditure on the project. This information can be used to get an estimate of the financial requirements of undertaking such a project in future. The amount of money spent per lane-mile can be used to plan out the fiscal requirements if a similar maintenance activity needs to be undertaken on a larger scale in future.

The summary report also shows the amount of money that needs to be paid to the personnel or for the equipment and material for that particular project. This can also help in resource allocation.

Other features of the application can be used for analytical purposes. The query function can be used to search for particular projects that dealt with a certain maintenance activity. This may be used to find out which crew members are experienced in conducting certain type of maintenance. Such queries may also be used to estimate the time required for completion of any particular maintenance activity. This can also be used to plan for the future.

The query or the summary report also provides information about the number of crew members involved in the project and the number of hours spent by each. Hence this can be used to plan crew schedules for similar projects in future. In the same manner the equipment and material requirements and schedules can also be planned out.

The application can also be used to view the details about any particular project. This helps in giving an idea of the various projects being undertaken and the resources allocated to each.
The information about the resource allocation and schedule for the different projects can be changed at any time. The change in a particular item will be reflected at all the places that the item is used. This helps in efficient and convenient maintenance of the pavement management activities.

The main uses of this application are either for an analysis of historical data or in planning for future projects. The planning is done using the information from the summary report. The query function also allows the supervisor or administrator to view and estimate the resources needed for future projects. The application is also very useful in maintaining an easily accessible database for the maintenance activities.
CHAPTER 7

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

A summary of the work done and the conclusions and recommendations for future work are presented in this chapter.

7.1 Summary

The road network is one of the most important elements of any town or city. It is the network of local streets, major roads and highways that integrate different parts and aspects of a society whether it be for linking the place of work to the place of residence or for providing medical help to a person. It would be impossible to think of having a society that does not have any use of a road network. The road network being such an important infrastructure element of any area of habitat, in general, it is critical that this important asset be maintained well.

The road network consists not only of streets and highways but also many other elements such as streetlights, pavement markers and markings, drainage facilities etc. These elements also need to be maintained in order to ensure public safety and welfare. The use of the road network leads to wear and tear and also damage that need to be repaired or maintained constantly. The deterioration of the elements may be due to several factors like traffic loading, crashes on the streets, environmental impacts etc.
Good maintenance of the road network is necessary to develop a more efficient transportation network.

The road network at most times forms an intricate mesh of elements and many a time maintenance activities might be occurring on quite a few of the elements. Keeping track of the maintenance activities on each element thus becomes a daunting task. Hence a management system that provides a methodology to keep track of these activities is required. With the advent of computers becoming common place in most offices and becoming a useful, easy to use computational tool, it is logical that computers be used to help keep track of the maintenance activities. Such a system is the computerized maintenance management system.

The development of a prototype roadway maintenance management system (MMS) has been presented in this thesis. The system is a windows based application that is recommended to be executed on a Pentium type processor with a minimum of 32MB of memory. The operating system should be MS-Windows based and MS-Access and MS Excel are also required. The system has a graphical user interface developed in Visual Basic 6.0 (VB) and all the information is stored in a Microsoft Access database.

Visual Basic has the provision to convert the program into an executable file and packaged along with all the necessary library files with a setup routine. This allows the program to be installed and run on any computer with the minimum specifications and without having to buy Visual Basic software.

The database is stored in Microsoft Access that comes as part of the Microsoft Office suite. Microsoft Office being common in most offices and work places, the
The roadway MMS developed in this thesis is simple and yet useful. The details about each project that is undertaken can be entered into the database through an easy to use interface. The interface has been built to minimize typing and hence the mistakes therein by providing drop-down boxes from which the value such as the crew member name can be chosen. At the same time if a new value needs to be entered then that can be done too with the drop-down box.

The program allows for addition, deletion, change, and analysis/query of the information about each project. There is also the provision for generating summary reports that give the total cost of the project along with a breakdown based on individual cost of personnel, equipment and material.

The application provides for more than one user of the system. Although only one can use the system at any given point of time, different users can be given access to the database by one person, designated as the administrator. The permission to access one or more of the tables is set and maintained by the administrator. This allows for some users to access the equipment table but not the personnel table.

The Roadway Maintenance Management System tool presented in this thesis has been developed as a simple and inexpensive decision support tool and with further improvements this tool can become an integral and useful part of any agency that has the responsibility of keeping track of maintenance activities.
7.2 Conclusions

The road network is a major infrastructure of any town or city. It is thus critical that the network be as well maintained as possible. The number of elements involved in a road network makes keeping track of all the activities a daunting task. However it is important to keep track of the projects being undertaken for correct and efficient fiscal management. Also the management of all the resources such as personnel, material and equipment is important for overall smooth operations.

The roadway maintenance management system developed here provides a simple and easy application that can be used to maintain a database with all the information for each project that is undertaken. The information consists of details about the three main resources, that is, personnel, equipment and material.

The system is inexpensive and may be purchased by small county or town offices and can be implemented with little or no training. The system provides a good means to keep track of the expenditure by providing summary reports and this may help the office in planning the allocation of funds for future projects.

The system also helps maintain historical data and also information about each individual resource. This might help the supervisors and managers in scheduling the resources.

The use of a computerized system may also lead to a reduction in the time spent on maintaining a database and reduce the paperwork involved. A computerized system is easy to update and change and thus can be maintained more easily.

A computerized can thus be used as a useful and efficient decision support tool help manage the public infrastructure better and provide a good life to the people of the
community. Certain limitations apply to the application developed in this thesis and the next section addresses these limitations and the work that may be done in future to overcome them.

7.3 Recommendations for Future Work

The tool may be improved by allowing for more than four elements of each resource to be entered for a given project. For example, at present maximum of four crew members can be entered for a project. However, there may be projects with more than four crew members involved.

The tool can be improved by allowing the user to customize the summary reports. A provision to customize the report in any way that the user wishes may involve a lot of work. It is suggested that some report generating software, such as Crystal Reports be interfaced with the application for this purpose.

Another avenue for improvement is by allowing the user to change or add fields to the database tables. This will require a lot of work on making the interface more versatile and dynamic in order to allow all the fields to be displayed in the input forms and other user interface windows. However, this vastly helps improve the application since each system and organization might have its own parameters that need to be taken into account.

It might also be helpful to have a statistical analysis tool that might provide graphs and charts of the expenditure or time management of each project or a group of projects. This will further help in evaluating the efficiency of the maintenance management system as a whole and the crew and management in particular.
A Geographic Information System (GIS) interface may also be added to display a map of the road network. The interface can be made interactive by allowing the user to choose a particular street segment or a facility on the road from the map and update information regarding the maintenance activities being carried out at that facility.

The improvements in technology and better and faster computer networks, are making the Internet more common place and more of a necessity in day to day life. The application tool may be improved by making the database and the program accessible over the Internet so that users can use the system from any remote computer system. This might require improving the database system and might also need implementation of database software that is suited to such remote and multiple user access.
APPENDIX A

A SAMPLE SUMMARY REPORT FROM THE CLARK COUNTY
PUBLIC WORKS DEPARTMENT
MAINTENANCE MANAGEMENT SYSTEM
Activity: Metro Callout

Work Order #8107787  FLAMINGO A RAINBOW, M OF  Date: 10/03/97  Completed: 10/03/97  File ID WA01
Project: Citizen Inquiry  From  Commission District: F  Entity: Spring Valley  TO  Firemap:  RFCID ID
Originator: HARPER  Foreman: FICKLIN  Assigned To: WOOLMAN  In House/Contract: I
Scheduled Start: 10/03/97  Scheduled Completion: 10/03/97  Estimated Qty  0.00 MHS *

Description of Work Completed

METRO REPORTED GLASS IN THE STREET, SWEPT BY C. WOOLMAN.

<table>
<thead>
<tr>
<th>RESOURCE</th>
<th>Labor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Func/Description</td>
<td>F/R</td>
</tr>
<tr>
<td>Street Sweeping</td>
<td>10/03/97</td>
</tr>
<tr>
<td>Function Total</td>
<td>1.00</td>
</tr>
<tr>
<td>Resource Total</td>
<td>1.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RESOURCE</th>
<th>Equipment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Func/Description</td>
<td>F/R</td>
</tr>
<tr>
<td>Street Sweeping</td>
<td>10/03/97</td>
</tr>
<tr>
<td>Function Total</td>
<td>1.00</td>
</tr>
<tr>
<td>Resource Total</td>
<td>1.00</td>
</tr>
<tr>
<td>Work Order Total</td>
<td>2.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Qty Cnt</th>
<th>UOM</th>
<th>Hours</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labor</td>
<td>1.00</td>
<td>$26.90</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Equipment</td>
<td>1.00</td>
<td>$42.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Material</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1.00</td>
<td>MHS</td>
<td></td>
<td>$68.90</td>
</tr>
</tbody>
</table>

* UOM average computed for all except workorder UOM's listed as LOT or DIST

Code for Premium Pay: B) Overtime, C) Call Back, D) Standby,
S) Comp Time (1 1/3), Q) Holiday Overtime, X) Shift Differential
Y) Shift Differential + Comp Time, Z) Shift Differential + Overtime

Generated by "A.S.K.N.O.W."
REFERENCES


94

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.


19) Norwich Technologies, Ontario, Canada:  


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
VITA

Graduate College
University of Nevada Las Vegas

Ranjit Menon

Address:
4209 Grove Circle, #2
Las Vegas, Nevada 89119.

Degrees:
Bachelor of Technology
Indian Institute of Technology Chennai
Chennai 600 036
TamilNadu
INDIA.

Publications:

Thesis Title:
A Windows Based Roadway Maintenance Management System

Thesis Examination Committee:
Chairperson: Dr. Shashi S. Nambisan, Ph.D., P.E.
Committee Member: Dr. Mohamed Kaseko, Ph.D.
Committee Member: Dr. Walter C. Vodrazka, Ph.D., P.E.
Graduate Faculty Representative: Dr. Kazem Taghva, Ph.D.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.